Determinants of Parent Unemployment in the United States in October of 2018

Laurel Urwick

2025-12-23

R Markdown

Note: CPS microdata are not included in this repository due to IPUMS data use restrictions.

Libraries, Data, Data Check

```
# libraries
library(dplyr)
library(ggplot2)
library(broom)
library(tidyr)
library(flextable)
library(officer)

# pull in data
cps <- read.csv("data/cps_oct2018.csv")
head(cps)
str(cps)
```

Create Variables and Binaries

```
# employed vs unemployed
cps <- cps[cps$EMPSTAT%in%c(10,12,20,21,22),]
cps$employed <- ifelse(cps$EMPSTAT%in%c(10,12),1,0)
table(cps$employed)
# 0 = unemployed, 1 = employed

# parent vs not
cps$parent <- ifelse(cps$YNGCH%in%0:17,1,0)
table(cps$parent)
# 0 = not a parent of a child under 18, 1 = parent

# parent only
parents <- cps[cps$parent==1,]

# single parent
parents$single_parent <- ifelse(parents$MARST%in%c(3,4,5,6),1,0)
table(parents$single_parent)
# 0 = parent with partner, 1 = single parent
```

```r
#cps single parent
cps$single_parent <- ifelse(cps$parent==1&cps$MARST%in%c(3,4,5,6),1,0)
table(cps$single_parent)
# 0 = not a single parent, 1 = single parent

# ages of children
parents$young_child <- ifelse(parents$NCHLT5>0,1,0)
table(parents$young_child)
# 0 = parent of older kid(s), 1 = parent of kid(s) under 5

# cps young children
cps$young_child <- ifelse(cps$parent==1&cps$NCHLT5>0,1,0)
table(cps$young_child)
# 0 = not parent of young children, 1 = parent of kid(s) under 5

# parents of multiples
parents$multiples <- ifelse(parents$NCHILD>=2,1,0)
table(parents$multiples)
# 0 = parent of single kid, 1 = parent of multiples

# cps multiples
cps$multiples <- ifelse(cps$parent==1&cps$NCHILD>=2,1,0)
table(cps$multiples)
# 0 = not parent of multiples, 1 = parent of multiples

# young parents
parents$age_at_first_child <- ifelse(parents$ELDCH%in%c(0,99),NA,parents$AGE-parents$ELDCH)
parents$early_parent <- ifelse(parents$age_at_first_child<=21,1,0)
table(parents$early_parent)
# 0 = parent after age 21, 1 = parent at or before 21

# cps young parents
cps$age_at_first_child <- ifelse(cps$parent==1&!(cps$ELDCH%in%c(0,99)),
                                 cps$AGE-cps$ELDCH,NA)
cps$early_parent <- ifelse(cps$parent==1&
                           !is.na(cps$age_at_first_child)&
                           cps$age_at_first_child<=21,1,0)
table(cps$early_parent)
```

Descriptive Statistics

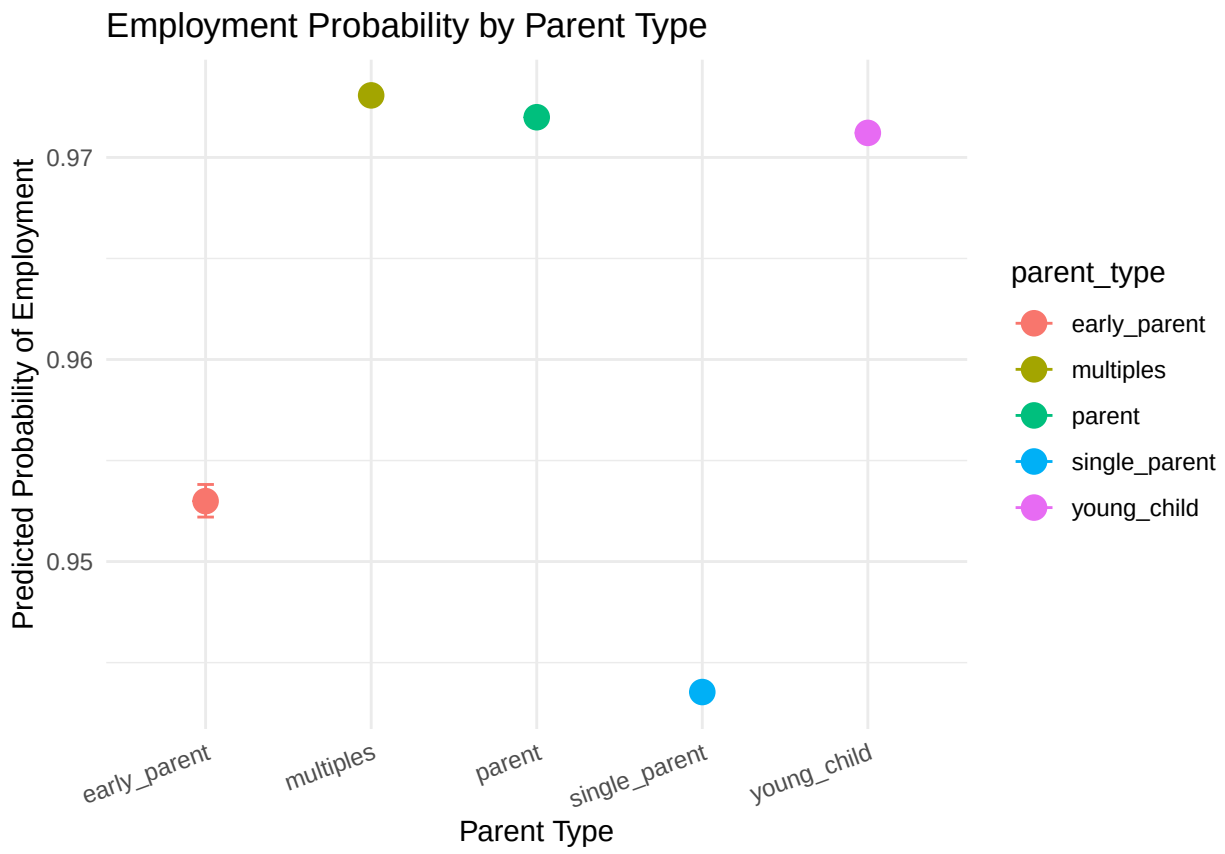
```r
# employment rate overall
prop.table(table(cps$employed))

# employment by parent status
prop.table(table(cps$employed,cps$parent),margin=2)

# employment by single-parent status
prop.table(table(parents$employed,parents$single_parent),margin=2)
```

Regression Models

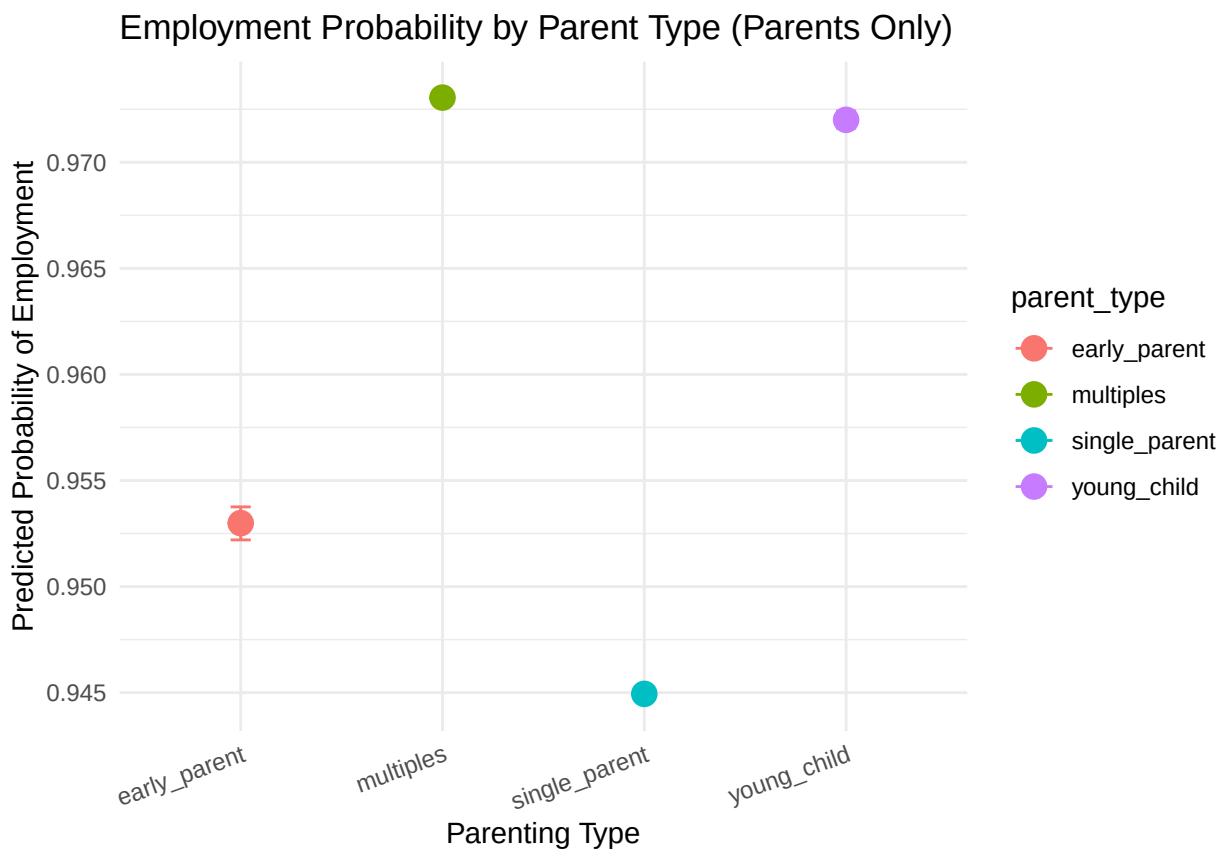
```r
# full-sample model
model_full <- glm(employed~parent+single_parent+young_child+multiples+early_parent,
                  data=cps,
                  family=binomial(link="logit"))
summary(model_full)

# parent-only model
model_parents <- glm(employed~single_parent+young_child+multiples+early_parent,
                     data=parents,
                     family=binomial(link="logit"))
summary(model_parents)
```

Graphs

```r
# full-sample model
cps$predicted_full <- predict(model_full,type="response")
plot_long <- cps%>%
  select(predicted_full,parent,single_parent,young_child,multiples,early_parent)%>%
  pivot_longer(cols=c(parent,single_parent,young_child,multiples,early_parent),
               names_to="parent_type",
               values_to="value")
plot_long <- plot_long%>%filter(value==1)
ggplot(plot_long,aes(x=parent_type,y=predicted_full,color=parent_type))+
  stat_summary(fun=mean,geom="point",size=4)+
  stat_summary(fun.data=mean_cl_boot,geom="errorbar",width=0.1)+
  labs(
    x="Parent Type",
    y="Predicted Probability of Employment",
    title="Employment Probability by Parent Type")+
  theme_minimal()+
  theme(axis.text.x=element_text(angle=20,hjust=1))
```




```
# parent-only model
parents$predicted_parent <- predict(model_parents,newdata=parents,type="response")
plot_long_parents <- parents%>%
  select(predicted_parent,single_parent,young_child,multiples,early_parent)%>%
  gather(key="parent_type",value="value",
         single_parent:early_parent)
plot_long_parents <- plot_long_parents%>%filter(value==1)
ggplot(plot_long_parents,
       aes(x=parent_type,y=predicted_parent,color=parent_type))+
  stat_summary(fun=mean,geom="point",size=4)+
  stat_summary(fun.data=mean_cl_boot,geom="errorbar",width=0.1)+
  labs(
    x="Parenting Type",
    y="Predicted Probability of Employment",
    title="Employment Probability by Parent Type (Parents Only)")+
  theme_minimal()+
  theme(axis.text.x=element_text(angle=20,hjust=1))
```



Tables for Paper

```
# table 1
var_meta <- tibble::tribble(
  ~dataset,  ~var,           ~label_0,                                        ~label_1,
  "cps",     "employed",     "0 = Unemployed",                                "1 = Employed",
  "cps",     "parent",       "0 = Not a parent of child < 18",                "1 = Parent of child < 18"
  "cps",     "single_parent","0 = Not a single parent",                       "1 = Single parent",
  "cps",     "young_child",  "0 = No child under 5",                          "1 = Has child under 5",
  "cps",     "multiples",    "0 = Not parent of multiples",                   "1 = Parent of 2+ children
  "cps",     "early_parent", "0 = First child after age 21 or never parent", "1 = First child before ag
  "parents", "single_parent","0 = Partnered parent",                          "1 = Single parent",
  "parents", "young_child",  "0 = Parent of only older child(ren)",           "1 = Parent of child(ren)
  "parents", "multiples",    "0 = Parent of one child",                       "1 = Parent of 2+ children
  "parents", "early_parent", "0 = First child after age 21",                  "1 = First child before ag
)
count_binary <- function(data, var_name) {
  v <- data[[var_name]]
  tibble(
    n0  = sum(v == 0, na.rm = TRUE),
    n1  = sum(v == 1, na.rm = TRUE),
    nNA = sum(is.na(v))
  )
}
# list of variables defined in cps and parents
vars_cps     <- c("employed", "parent", "single_parent", "young_child", "multiples", "early_parent")
```

```r
vars_parents <- c("single_parent", "young_child", "multiples", "early_parent")

# cps counts
cps_counts <- purrr::map_dfr(
  vars_cps,
  ~ count_binary(cps, .x) %>% mutate(dataset = "cps", var = .x)
)

# parents counts
parents_counts <- purrr::map_dfr(
  vars_parents,
  ~ count_binary(parents, .x) %>% mutate(dataset = "parents", var = .x)
)

var_counts <- bind_rows(cps_counts, parents_counts)
var_table <- var_meta %>%
  left_join(var_counts, by = c("dataset", "var")) %>%
  select(dataset, var, label_0, n0, label_1, n1, nNA)

var_table

# table 2 - logistic regression estimates (full sample): employment and parenting
table_full_clean <- tidy(model_full) %>%
  filter(term != "(Intercept)") %>%
  mutate(
    term = recode(term,
                  parent        = "Parent",
                  single_parent = "Single parent",
                  young_child   = "Child under age 5",
                  multiples     = "Two or more children",
                  early_parent  = "First child before age 22"
    ),
    stars = case_when(
      p.value < 0.001 ~ "***",
      p.value < 0.01  ~ "**",
      p.value < 0.05  ~ "*",
      TRUE            ~ ""
    ),
    estimate_r = round(estimate, 3),
    se_r       = round(std.error, 3),
    `Estimate (SE)` = paste0(estimate_r, stars, " (", se_r, ")")
  ) %>%
  select(Variable = term, `Estimate (SE)`)
table_full_clean

# table 3 - logistic regression estimates (parents only): employment and parenting
table_parents_clean <- tidy(model_parents) %>%
  filter(term != "(Intercept)") %>%
  mutate(
    term = recode(term,
                  single_parent = "Single parent",
                  young_child   = "Child under age 5",
                  multiples     = "Two or more children",
```

```
                  early_parent  = "First child before age 22"
    ),
    stars = case_when(
      p.value < 0.001 ~ "***",
      p.value < 0.01  ~ "**",
      p.value < 0.05  ~ "*",
      TRUE            ~ ""
    ),
    estimate_r = round(estimate, 3),
    se_r       = round(std.error, 3),
    `Estimate (SE)` = paste0(estimate_r, stars, " (", se_r, ")")
  ) %>%
  select(Variable = term, `Estimate (SE)`)
table_parents_clean
```